Capstone Proposal

Marcel Stolin
marcelstolin@gmail.com

Abstract

The Udacity machine learning engineer nanodegree is an online course where students can learn advanced machine learning algorithms and how to package and deploy models to a production environment. It has been made to gain practical knowledge [4].
In this proposal I will explain the Dog Breed Image Classification capstone project, the problem, what data will be used, how the model will be evaluated and the project design.

1 Domain Background

Image Classification is an active research field in multimedia. Applications of image classification are widely used in security, healthcare, entertainment and many more.
This project is mostly created for entertainment but there are serious research projects made about image classification with animals. For example Trnovszky, Kamencay, Orjesek, Benco and Sykora published a paper where they created a Convolutional Neural Network (CNN) to propose the animal species from an images of an animal [7].

2 Problem Statement

The goal of this project is, to use a CNN to predict the dog from an image. If there is no dog on the image, but a human, then return the resembling dog breed according to the human. Otherwise, if a dog is detected, return the predicted dog breed. Therefore we face the following main issues in this project.

1. Create a model to detect a human
2. Create a model to predict the dog breed
3. Create an algorithm, which will return the predicted dog breed for the image

We will develop at least two different models: A dog breed classifier and a human face classifier. For that we can develop a CNN from scratch or use pre-trained models for this specific case, like VGG-16 [6] or ResNet-50 [5].

3 Datasets and Inputs

The data for this project was selected by Udacity. For detecting dog breeds we will use a custom dataset provided by Udacity. This dataset is already separated into a train, test and valid folder. We will use the images according to their purpose, the train images will be used to train the model, the test images to test/evaluate the model and the valid dataset to test the algorithm. Each dog images are labeled with the according dog breed.
For detecting humans we will use the LFW (Labeled Face in the Wild) [2] dataset from the University of Massachusetts. The LFW dataset consists of face photographs. Each face has been labeled with the name of the person.

4 Solution Statement

First we need to come up with a way to detect human faces. For this issue, OpenCV provides the Haar cascade classifier [8]. The classifier is trained from images with an machine learning approach based on a cascade function, to detect objects in images [1].
As mentioned in 2 we will develop a CNN from scratch. The CNN will train a model to classify dog breeds from images. We will also use the VGG-16 model to detect dogs in images. For transfer learning we will use the ResNet-50 model which has won several prices on image classification tasks and therefore, it is a good fit for this case. We will aim an accuracy of at least 60% to be sure our model is well trained.

5 Benchmark Model

As mentioned in 4 we will use the ResNet-50 model against our scratch model. The ResNet-50 model uses images of 224x224 pixels. Therefore we will resize our dog breed images to 224x224 pixels.

6 Evaluation Metrics

In cases of classification, we can use the accuracy, F1-Score, precision and recall metrics to evaluate our models.

6.1 Confusion Matrix

It is easy to use a confusion matrix to visualise the results of a classifier.
Given the confusion matrix, we can calculate the metrics mentioned above.

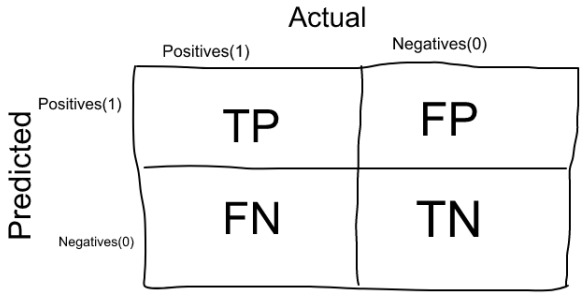

Figure 1: Confusion Matrix [3]

6.2 Accuracy

Accuracy is the number of correct predictions made by the model over all predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

6.3 Precision

Precision is measure which tells us, what proportion of predicted positives are truly positives.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

6.4 Recall

The recall metric tells us, what proportion of actual positives are correctly classified.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

6.5 F1-Score

The F1-Score is the harmonic mean of precision and recall.

$$F1Score = 2 * Precision * \frac{Recall}{Precision + Recall} \tag{4}$$

7 Project Design

Udacity provides a Jupyter Notebook file where each step is described and highlighted. The notebook consist of the following sections:

1. Import Datasets

2. Detect Humans
3. Detect Dogs
4. Create a CNN to Classify Dog Breeds
5. Create a CNN to Classify Dog Breeds (using Transfer Learning)
6. Write your Algorithm
7. Test Your Algorithm

References

[1] Cascade classifier. https://docs.opencv.org/.

[2] Labeled face in the wild. http://vis-www.cs.umass.edu/lfw.

[3] Performance metrics for classification problems in machine learning. https://medium.com/thalus-ai/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b.

[4] Udacity machine learning engineer nanodegree. https://www.udacity.com/course/machine-learning-engineer-nanodegree–nd009t.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. pages 770–778, June 2016.

[6] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014.

[7] Tibor Trnovszky, Patrik Kamencay, Richard Orješek, Miroslav Benco, and Peter Sykora. Animal recognition system based on convolutional neural network. *Advances in Electrical and Electronic Engineering*, 15, 10 2017.

[8] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. pages 511–518, 2001.